Algorithms for Programming Contests - Week 7

Stefan Jaax, Philipp Meyer, Christian Müller conpra@in.tum.de

22/06/2016

Greedy Algorithms

Definition (Greedy Heuristic)

Make locally optimal choice at each step.

- Simple, Easy
- Not dependent on future choices
- Not dependent on solutions to created subproblems
- Never reconsiders choices
- Short-sighted: may be inaccurate or incomplete
- in general: no guarantee

Greedy: Examples for optimal approaches

- *Prim/Kruskal* for MST.
- *Dijkstra* for SSSP.

Greedy: Suboptimal Approximation

Definition (0-1 Knapsack Problem)

Given

- a knapsack with maximum weight capacity W,
- n items with weights w_i and values v_i, each.

Pick items so that the total sum of their values v_i is maximized, subject to the constraint that the total sum of weights w_i does not exceed the capacity W.

0-1 Knapsack

Greedy approach: Sort items by v_i/w_i in decreasing order and put items into knapsack, in that order, until weight capacity is reached.

Approach is suboptimal

0/1-Knapsack instance with two items where $(w_1, v_1) = (W, W-1)$ and $(w_2, v_2) = (1, 1)$.
$\Longrightarrow v_1/w_1 < v_2/w_2$. Thus Greedy always picks second item, even though first one is optimal. If W goes to infinity, the greedy solution becomes arbitrarily bad.

Modified Greedy

Compute greedy solution. Then consider most profitable item. If item is a better choice than greedy solution, return item, otherwise return greedy solution.
$\Longrightarrow$ Yields a 2-approximation.

Fractional Knapsack

In *Fractional Knapsack* items need not be picked in whole pieces, but fractions $0 < f_i < 1$ of items are allowed.
$\Longrightarrow$ Greedy approach is optimal for Fractional Knapsack.

General Problem Statement

General setting

Given a set of objects X and a function $f\colon 2^X \to \mathbb{R}$. Find $S \subseteq X$ that maximizes/minimizes f.

- In general: Intractable / check all $2^{|X|}$ subsets.
- Need restrictions on S and f.

A sufficient criterion for optimality

Definition (Independence system)

Let X be a finite set. An independence system over X is a family J of subsets of X with the following two properties

1. J is non-empty.
2. If $I \in J$ and $I' \subseteq I$, then $I' \in J$.

Definition (Matroid)

A matroid is an independence system (X, J), satisfying the additional property:
If A, B are in J with $|A| = |B| + 1$, then there is an element $a \in A \setminus B$ such that $B \cup \{a\} \in I$.

Independent Sets

Definition (Independent sets)

Let (X, J) be a matroid. We call elements of J *independent sets*.
The largest elements of J are called *maximally independent sets*.

The size of maximally independent sets is unique for every matroid!

Matroid-Greedy-Theorem

Theorem (Greedy is optimal for matroids)

Let $\mathcal{M} = (X, J)$ be an independence system and let f be a non-negative cost function on X. Suppose the greedy algorithm iteratively adds the cheapest element from X that maintains independence (i.e. the resulting set must be an independent set).
Then the greedy algorithm produces a maximally independent set of minimal total cost among all maximally independent sets for all non-negative cost functions f if and only if $\mathcal{M}$ is a matroid.